A network-oriented adaptive agent model for learning regulation of a highly sensitive person's response

Tran, Linh; Treur, Jan; Tuinhof, Denice J.

published in
Advances in Practical Applications of Agents, Multi-Agent Systems, and Complexity
2018

DOI (link to publisher)
10.1007/978-3-319-94580-4_20

document version
Publisher's PDF, also known as Version of record

A Network-Oriented Adaptive Agent Model for Learning Regulation of a Highly Sensitive Person's Response

Linh Tran, Jan Treur(✉), and Denice J. Tuinhof

Behavioural Informatics Group, Vrije Universiteit Amsterdam, Amsterdam, The Netherlands
trplinh@gmail.com, j.treur@vu.nl, denicetuinhof@gmail.com

Abstract. Inspired by the work of Elaine Aron, in this paper a human-like adaptive computational agent model of the internal processes of a highly sensitive person (HSP) is presented. This agent model was used to get a better understanding of what goes wrong in these internal processes once this person gets upset. A scenario is addressed where a highly sensitive person will get upset by an external stimulus and will not be able to calm down by herself. Yet in a social context the interaction with a second person (without high sensitivity) will calm the HSP down, thus contributing to regulation. To obtain an adaptive model a Hebbian learning connection was integrated. During interaction with a second person this Hebbian learning link will become stronger, which makes it possible for a HSP to become independent after some time and be able to regulate upsetting external stimuli all by herself.

Keywords: Highly sensitive person · Hebbian learning
Sensory processing sensitivity

1 Introduction

According to Jagiellowicz *et al*. [1] one fifth of the population has high Sensory-Processing Sensitivity also called (HSP); see also [2, 4, 6, 8, 9]. All these people would profit from more knowledge on their disorder. So, as a lot of people suffer from HSP and still a lot remains unknown, there is a demand for more insight. Therefore, work in this research area is challenging but interesting and would be a good contribution to society. If the internal processes of HSP can be modelled computationally, our understanding of the underlying processes will grow. This enlarged knowledge will support us in the interaction and therapy of highly sensitive persons.

To obtain a human-like computational adaptive agent model, a couple of design decisions were made. First of all, a woman was chosen to be our sensitive person. After reading [5, 7] it was more logical to pick a woman because they are more extreme on the high sensitive person scale, which also includes the reaction on internal and external stimuli. In addition, a negative stimulus was chosen with visual elements (e.g., a flashing light), so that the person may avoid the stimulus by changing her gaze, which

Y. Demazeau et al. (Eds.): PAAMS 2018, LNAI 10978, pp. 248–261, 2018.
https://doi.org/10.1007/978-3-319-94580-4_20

is observable by other persons. Furthermore, a learning aspect was incorporated making the model adaptive. This will give insights in how to make a HSP more independent in upsetting situations. More insight in the learning of a HSP can contribute to therapy for HSP, for example, via a virtual training environment using the model during the education of therapists.

To obtain the human-like agent model, it was decided to use a Network-Oriented Modeling approach as described in [14]. This Network-Oriented approach to agent modeling can be viewed as standing on the one hand in the causal modeling tradition in AI (e.g., [16–18]), and on the other hand in the perspective on mental states and their causal relations in Philosophy of Mind (e.g., [14]). It adds dynamics to causal relations by using an additional temporal dimension, and describes a causal model as a temporal-causal network model. In the temporal-causal network models used, states change over time due to the causal impacts they have on each other, but also the causal relations can change, thus enabling an adaptive network model. In this way learning can be incorporated; for example, Hebbian learning is incorporated by modeling that a connection between two states becomes stronger when both states are active simultaneously.

In the paper in Sect. 2 some background knowledge from Neuroscience is discussed. Next, in Sect. 3 the adaptive agent model is introduced. In Sect. 4 some example simulations are presented. Section 5 discusses the role of requirements and parameter tuning, and Sect. 6 discusses the Mathematical Analysis performed to verify whether the implemented model does what is expected. Finally, Sect. 7 is a discussion.

2 Background Knowledge

This section briefly discusses background knowledge as found in literature such as [1, 5, 7, 9, 10]. A highly sensitive person, or a person with Enhanced Sensory-Processing Sensitivity, is a person which is characterized as being really sensitive to external and internal stimuli, intense emotions and with a preference for deep processing of information [1, 2]. In this research the focus is on the sensitivity for external stimuli [4]. Intense stimuli that may upset a HSP are bright lights, strong smells, coarse fabrics or sirens [4]. Not only in expressions are HSP different from non-HSP. HSP process sensory information much more deeply due to biological differences in their nervous system [7, 8]. These differences and the level of high sensitivity can be measured on a scale. In 1997 Aron and Aron developed a highly sensitive person scale [9]. The HSP scale is a measure of sensory-processing sensitivity, which is conceptualized as involving both high levels of sensitivity to subtle stimuli and being easily over-aroused by external stimuli [10]. This research concludes that men are more moderate on this scale then woman. Which makes in the scenario addressed below a woman more logical as the HSP.

Hebbian learning is a learning mechanism in the (human) brain that suggests that *'units that fire together, wire together'*. According to Hebb the 'efficiency' of a given neuron, in contributing to the firing of another, could increase as that cell is repeatedly involved in the activation of the second [11]. Therefore, neurons that have correlated firing will strengthen their mutual interaction and increase the strength of synapses they form with each other. Certainly, these mutual interactions of neurons do not arise spontaneously. It could take days, weeks or even years for a Hebbian-like mechanism to provide a strong connection. This idea is consistent with the neural mechanisms of long-term potentiation (LTP) and long-term depression (LTD) [12]. LTP increases synaptic efficacy, while LTD weakens the synaptic efficacy. Therefore, LTP and LTD influence the efficacy of synapses or junctions between neurons. Subsequently, the extent to which activity in a sending neuron leads to depolarization of a receiving neuron is influenced as well. While these are long-term processes, it is suggested that Hebbian learning models are highly relevant for investigating development. Moreover, Hebbian learning models can account for a wide range of behaviors and changes during development [12]. Thus, a Hebbian learning model is an eligible addition to the model of a highly sensitive person changing her behaviour over time as addressed below.

The scenario addressed is as follows. After a siren (existing of both light and sound) goes off in the room, this immediately has an effect on Arnie. Arnie is a highly sensitive female and does not like loud noises and bright lights. When Arnie sees the lights, her body language adjusts to the presence of this, in addition she tries to cope with the siren by trying to avoid it with her gaze and speaking her discomfort out loud. Therefore, she communicates this to her friend Bert (a person with normal sensitivity) that the presence of the siren makes her feel uncomfortable. Bert also senses that Arnie's body state is uncomfortable. In addition, Bert notices that Arnie is avoiding the siren with her gaze. These three inputs make Bert want to undertake action and comfort his friend. Bert tries to regulate the situation and prepares some comforting words. He expresses these comforting words to Arnie. Because Arnie avoided the siren by adjusting her gaze, her body is already a little bit less stressed. When her friend Bert supports her with some comforting words, her body relaxes. Arnie needed Bert to comfort her because her internal regulation process is not strong enough to regulate her own internal and body state. However, due to Bert's comforting words, Arnie is learning how to comfort herself in the same way. Through Hebbian learning, Arnie is learning to cope with an upsetting external stimulus on her own a bit better.

3 The Network-Oriented Adaptive Agent Model

First the Network-Oriented Modeling approach used to model the adaptive agent is briefly explained. As discussed in detail in Chap. 2 of [14], this approach is based on temporal-causal network models which can be represented at two levels: by a conceptual representation and by a numerical representation. These model representations can be used not only to display graphical network pictures, but also for numerical simulation. Furthermore, they can be analyzed mathematically and validated by comparing their simulation results to empirical data. They usually include a number of parameters for domain, person, or social context-specific characteristics. A conceptual representation of a temporal-causal network model in the first place involves representing in a declarative manner states and connections between them that represent (causal) impacts of states on each other, as assumed to hold for the application domain addressed. The states are assumed to have (activation) levels that vary over time. In reality, not all causal relations are equally strong, so some notion of *strength of a* *connection* is used. Furthermore, when more than one causal relation affects a state, some way to *aggregate multiple causal impacts* on a state is used. Moreover, a notion of *speed of change* of a state is used for timing of processes. These three notions are covered by elements in the Network-Oriented Modelling approach based on temporal-causal networks, and form the defining part of a conceptual representation of a specific temporal-causal network model:

- **Strength of a connection $\boldsymbol{\omega}_{X,Y}$**
  Each connection from a state X to a state Y has a *connection weight value* $\boldsymbol{\omega}_{X,Y}$ representing the strength of the connection, often between 0 and 1, but sometimes also below 0 (negative effect) or above 1.
- **Combining multiple impacts on a state $\mathbf{c}_Y$(..)**
  For each state (a reference to) a *combination function* $\mathbf{c}_Y$**(..)** is chosen to combine the causal impacts of other states on state Y.
- **Speed of change of a state $\boldsymbol{\eta}_Y$**
  For each state Y a *speed factor* $\boldsymbol{\eta}_Y$ is used to represent how fast a state is changing upon causal impact.

Combination functions can have different forms, as there are many different approaches possible to address the issue of combining multiple impacts. The applicability of a specific combination rule for this may depend much on the type of application addressed, and even on the type of states within an application. Therefore, the Network-Oriented Modelling approach based on temporal-causal networks incorporates for each state, as a kind of label or parameter, a way to specify how multiple causal impacts on this state are aggregated. For this aggregation a number of standard combination functions are made available as options and a number of desirable properties of such combination functions have been identified (see [15], Chap. 2, Sects. 2.6 and 2.7), some of which are the scaled sum function and the advanced logistic sum function:

$$\mathbf{ssum}_\lambda(V_1, \ldots, V_k) = (V_1 + \ldots + V_k)/\lambda \quad \text{with}\, \lambda > 0$$

$$\mathbf{alogistic}_{\sigma,\tau}(V_1, \ldots, V_k) = [(1/(1 + \mathbf{e}^{-\sigma(V_1 + .. + V_k - \tau)}) - (1/(1 + \mathbf{e}^{\sigma\tau}))](1 + \mathbf{e}^{-\sigma\tau}) \\ \text{with}\, \sigma,\ \tau \geq 0$$

The above three concepts (connection weight, speed factor, combination function) can be considered as parameters or labels representing characteristics in a network model. In Fig. 1 a graphical conceptual representation of the temporal-causal network model presented here is shown. Adding the three above-mentioned types of labels to such a picture as shown in Fig. 1 provides a labelled graph as the model specification. In a non-adaptive network model these parameters are fixed over time. But to model processes by adaptive networks, not only the state levels, but also the values of some of these parameters can change over time.

The explanation of the depicted states can be found in Table 1. The two persons involved are indicated by the dotted boxes: A is the considered female Arnie with HSP, and B a normal person Bert. The states not in the boxes are world states, $\mathrm{ws}_{avoiding,s}$ one for Arnie's avoiding gaze, ws_s one for the stimulus s and ws_b one for the body state b. The gaze state has effect on what A represents, and (through observation) it is represented by B. The body state b of A is represented by both. In this way the two persons interact via the world states, in addition to communication between them, which is modeled by the connections from $esc_{B,b}$ to $rsc_{A,B}$, and from $esc_{A,n}$ to $rsc_{B,A}$.

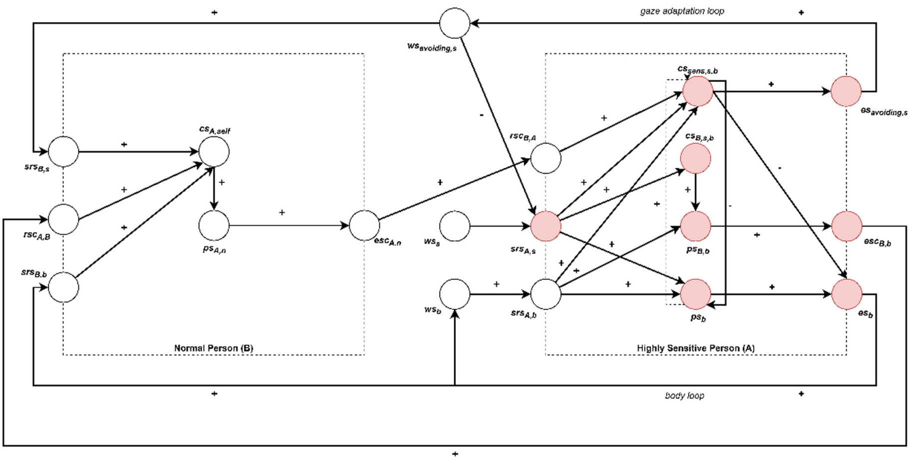

Fig. 1. Graphical conceptual representation of the network-oriented agent model.

Table 1. Overview of the states used in the model and their explanation

State	Explanation
$srs_{A,s}$	Sensory representation state of stimulus s by A
$srs_{B,s}$	Sensory representation state of avoiding gaze of A by B
$srs_{X,b}$	Sensory representation state of body state b by X (= A or B)
$rsc_{B,A}$	Representation state of A for communication from B
$rsc_{A,B}$	Representation state of B for communication from A
ps_b	Preparation state for body state b by A
$ps_{B,b}$	Preparation state for communication b of person A to person B
$ps_{A,n}$	Preparation state for communication n of person B to person A
$cs_{B,s,b}$	Control state for preparation state for communication by A of b to B
$cs_{A,self}$	Control state for self-other distinction of B concerning A
$cs_{sens,s}$	Control state for enhanced sensory sensitivity for s
es_b	Execution state for body state b
$esc_{B,b}$	Execution state for communication of body state b to B
$esc_{A,n}$	Execution state for communication of n to A
$es_{avoiding,s}$	Execution state for avoidance of stimulus s
ws_s	World state for stimulus s
ws_b	World state for body state b
$ws_{avoiding,s}$	World state for gaze avoiding stimulus s

The impact from one state on another state can have different values, positive or negative, usually in the interval [−1, 1]. These are some of the impacts:

- The presence ws_s of the stimulus has a positive influence on the sensory representation state presentation $\mathrm{srs}_{A,s}$ of the stimulus by Arnie.
- When Arnie has representation $\mathrm{srs}_{A,s}$ active, the control state $\mathrm{cs}_{sens.s,b}$ of the body state of Arnie for enhanced sensory sensitivity of the stimulus gets a positive impact, as well as the control state $\mathrm{cs}_{B,s,b}$ for body state for self-other distinction concerning Bert and the preparation state ps_b for the body state of Arnie get a positive impact.
- The body control state $\mathrm{cs}_{sens,s,b}$ of Arnie (attempting to regulate the enhanced sensory sensitivity), will stimulate the action $\mathrm{es}_{avoiding,s}$ of avoiding the stimulus with her gaze and will have a positive influence of the expression es_b of Arnie's body state.
- The body control state $\mathrm{cs}_{sens,s,b}$ of Arnie also has a negative influence on the body state expression es_b of Arnie, and also on the preparation ps_b; because Arnie is a sensitive person, Arnie has difficulties to suppress her body state expression, so this negative impact may be too weak.
- The control state $\mathrm{cs}_{B,s,b}$ for self-other distinction will stimulate the preparation $\mathrm{ps}_{B,b}$ for communication of the body state of Arnie to Bert.
- The preparation ps_b of the body state of Arnie will have a positive effect on the expression es_b of the body state of Arnie.
- The avoidance $\mathrm{es}_{avoiding,s}$ of Arnie of his gaze to the stimulus, will have a positive effect on the world state $\mathrm{ws}_{avoiding,s}$ of avoiding the stimulus with her gaze.
- The world state $\mathrm{ws}_{avoiding,s}$ where Arnie avoids the siren with her gaze will have a suppressing influence on the sensory representation srs_s that Arnie receives of the stimulus.
- The dotted arrow from ps_b to $\mathrm{cs}_{sens,s,b}$ indicates the connection on which Hebbian learning is applied. This connection may not be very strong initially, which makes the control state $\mathrm{cs}_{sens,s,b}$ not very active in case of high levels of ps_b (emotional response on the stimulus). By learning, the connection can become stronger which enables more effective control (by suppressing them stronger) of the emotional responses ps_b and es_b on the stimulus.

A conceptual representation of a temporal-causal network model can be transformed in a systematic or even automated standard manner into an equivalent numerical representation of the model as follows [14], Chap. 2:

- at each time point t each state Y in the model has a real number value in the interval [0, 1], denoted by $Y(t)$
- at each time point t each state X connected to state Y has an impact on Y defined as $\mathbf{impact}_{X,Y}(t) = \boldsymbol{\omega}_{X,Y}\, X(t)$ where $\boldsymbol{\omega}_{X,Y}$ is the weight of the connection from X to Y
- The *aggregated impact* of multiple states X_i on Y at t is determined using a *combination function* $\mathbf{c}_Y(..)$:

$$\mathbf{aggimpact}_Y(t) = \mathbf{c}_Y(\mathbf{impact}_{X_1,Y}(t), \ldots, \mathbf{impact}_{X_K,Y}(t)) \\ = \boldsymbol{c}_Y(\boldsymbol{\omega}_{X_1,Y}\boldsymbol{X_1}(\boldsymbol{t}), \ldots, \boldsymbol{\omega}_{X_K,Y}\boldsymbol{X_k}(\boldsymbol{t}))$$

where X_i are the states with connections to state Y

- The effect of $\mathbf{aggimpact}_Y(t)$ on Y is exerted over time gradually, depending on speed factor η_Y:

$$Y(t + \Delta t) = Y(t) + \boldsymbol{\eta}_Y[\mathbf{aggimpact}_Y(t) - Y(t)]\Delta t$$

or

$$\mathbf{d}Y(t)/\mathbf{d}t = \boldsymbol{\eta}_Y[\mathbf{aggimpactY}(\mathbf{t}) - Y(t)]$$

- Thus, the following *difference* and *differential equation* for Y are obtained:

$$Y(t + \Delta t) = Y(t) + \boldsymbol{\eta}_Y[\mathbf{c}_Y(\boldsymbol{\omega}_{X_1,Y}X_1(t), \ldots, \boldsymbol{\omega}_{X_k,Y}X_k(t)) - Y(t)]\,\Delta t$$

$$\mathbf{d}Y(t)/\mathbf{d}t = \boldsymbol{\eta}_Y[\mathbf{c}_Y(\boldsymbol{\omega}_{X_1,Y}X_1(t), \ldots, \boldsymbol{\omega}_{X_k,Y}X_k(t)) - Y(t)]$$

As explained above in a general setting, from the conceptual representation of a temporal-causal network model (including labels for connection weights $\boldsymbol{\omega}_{X,Y}$, combination functions $\mathbf{c}_Y(\ldots)$ and speed factors $\boldsymbol{\eta}_Y$), the difference and differential equations of the numerical representation can be easily generated. For the key states the difference equations are:

- For any point in time t for state $\mathrm{srs}_{A,s}$

$$\mathrm{srs}_{A,s}(t + \Delta t) = \mathrm{srs}_{A,s}(t) + \eta_{\mathrm{srs}_s}[\mathrm{c}_{\mathrm{srs}_s}(\omega_{\mathrm{ws}_s,\mathrm{srs}_s}\mathrm{ws}_s(t)) - \mathrm{srs}_{A,s}(t)]\Delta t$$

- For any point in time t for state $\mathrm{cs}_{sens,s}$

$$\mathrm{cs}_{sens,s,b}(t + \Delta t) = \mathrm{cs}_{sens,s,b}(t) + \eta_{\mathrm{cs}_{sens,s,b}}\,[\mathrm{c}_{\mathrm{cs}_{sens,s,b}}(\omega_{\mathrm{rsc}_{B,A},\mathrm{cs}_{sens,s,b}}\mathrm{rsc}_{sens,s}(t), \\ \omega_{\mathrm{srs}_{A,S},\mathrm{cs}_{sens,s,b}}\mathrm{srs}_{A,s}(t), \omega_{\mathrm{srs}_{A,b},\mathrm{cs}_{sens,s,b}}\mathrm{srs}_{A,b}(t), \omega_{\mathrm{ps}_b,\mathrm{cs}_{sens,s,b}}\mathrm{ps}_b(t)) - \mathrm{cs}_{sens,s,b}(t)]\,\Delta t$$

- For any point in time t for state $\mathrm{cs}_{B,s,b}$

$$\mathrm{cs}_{B,s,b}(t + \Delta t) = \mathrm{cs}_{B,s,b}(t) + \eta_{\mathrm{cs}_{B,s,b}}[\mathrm{c}_{\mathrm{srs}_s}(\omega_{\mathrm{srs}_{A,s},\mathrm{cs}_{B,s,b}}\mathrm{srs}_{A,s}(t)) - \mathrm{cs}_{B,s,b}(t)]\Delta t$$

- For any point in time t for state $\mathrm{ps}_{B,b}$

$$\mathrm{ps}_{B,b}(t + \Delta t) = \mathrm{ps}_{B,b}(t) + \eta_{\mathrm{ps}_{B,b}}[\mathrm{c}_{\mathrm{srs}_s}(\omega_{\mathrm{srs}_{A,b},\mathrm{ps}_{B,b}}\mathrm{srs}_{A,b}(t), \omega_{\mathrm{cs}_{B,s,b},\mathrm{ps}_{B,b}}\mathrm{cs}_{B,s,b}(t)) \\ - \mathrm{ps}_{B,b}(t)]\Delta t$$

- For any point in time t for state ps_b

$$\mathrm{ps}_b(t+\Delta t) = \mathrm{ps}_b(t) + \eta_{\mathrm{ps}_b}[\mathrm{c}_{\mathrm{srs}s}(\omega_{\mathrm{srs}_{A,b},\mathrm{ps}_b}\mathrm{srs}_{A,b}(t), \omega_{\mathrm{srs}_{A,s},\mathrm{ps}_b}\mathrm{srs}_{A,s}(t), \omega_{\mathrm{cs}_{sens,s},\mathrm{ps}_b}\mathrm{cs}_{sens,s}(t)) - \mathrm{ps}_b(t)]\Delta t$$

- For any point in time t for state $\mathrm{es}_{avoiding,s}$

$$\mathrm{es}_{avoiding,s}(t+\Delta t) = \mathrm{es}_{avoiding,s}(t) + \eta_{\mathrm{es}_{avoiding,s}}[\mathrm{c}_{\mathrm{es}_{avoiding,s}}(\omega_{\mathrm{cs}_{sens,s,b},\mathrm{es}_{avoiding,s}}\mathrm{cs}_{sens,s,b}(t)) - \mathrm{es}_{avoiding,s}(t)]\Delta t$$

- For any point in time t for state $\mathrm{esc}_{B,b}$

$$\mathrm{esc}_{B,b}(t+\Delta t) = \mathrm{esc}_{B,b}(t) + \eta_{\mathrm{esc}_{B,b}}[\mathrm{c}_{\mathrm{esc}_{B,b}}(\omega_{\mathrm{ps}_{B,b},\mathrm{esc}_{B,b}}\mathrm{ps}_{B,b}(t)) - \mathrm{esc}_{B,b}(t)]\,\Delta t$$

- For any point in time t for state es_b

$$\mathrm{es}_b(t+\Delta t) = \mathrm{es}_b(t) + \eta_{\mathrm{es}_b}[\mathrm{c}_{\mathrm{esc}_{B,b}}(\omega_{\mathrm{cs}_{sens,s,b},\mathrm{es}_b}\mathrm{cs}_{sens,s,b}(t), \omega_{\mathrm{ps}_b,\mathrm{es}_b}\mathrm{ps}_b(t)) - \mathrm{es}_b(t)]\Delta t$$

The weight $\omega = \omega_{\mathrm{ps}b,\mathrm{cs}sens,s}$ of the connection from ps_b to $\mathrm{cs}_{sens,s}$ to which Hebbian learning is applied is dynamic and can be handled as if it is a state; the following difference equation is used:

$$\omega(t+\Delta t) = \omega(t) + \eta_\omega[\mathrm{c}_\omega(\mathrm{ps}_b(t), \mathrm{cs}_{sens,s,b}(t), \omega(t)) - \omega(t)]\,\Delta t$$

with speed factor (learning rate) η_ω, and combination function

$$\mathrm{c}_\omega(V_1, V_2, W) = V_1 V_2 (1 - W) + \mu\, W$$

Here μ is a persistence parameter with values between 0 and 1, where $\mu = 1$ means fully persistent, and $\mu < 1$ indicates some extent of extinction. So,

$$\mathrm{c}_\omega(\mathrm{ps}_b(t), \mathrm{cs}_{sens,s,b}(t), \omega(t)) = \mathrm{ps}_b(t)\,\mathrm{cs}_{sens,s,b}(t)\,(1 - \omega(t)) + \mu\,\omega(t)$$

and

$$\omega(t+\Delta t) = \omega(t) + \eta_\omega[\mathrm{ps}_b(t)\mathrm{cs}_{sens,s,b}(t)\,(1 - \omega(t)) - (1-\mu)\,\omega(t)]\,\Delta t$$

4 Simulation Results

An estimation was made about which choices for the parameters (connection weights, combination functions, speed factors) to use, and initial values for the states were chosen. The connection weights can be found in the Table 2. All speed factors were 0.25. For all states with a single incoming connection the identity combination function has been used: $\mathbf{id}(V) = V$. For states $\mathrm{ps}_{B,b}$ and $\mathrm{cs}_{A,self}$ the scaled sum combination function has been used with scaling factor 2 and 3, respectively. For states $\mathrm{srs}_{A,s}$, ps_b,

$cs_{sens,s,b}$ and es_b the advanced logistic combination function has been used with steepness and threshold combinations (σ, τ) = (4.137, 0.139), (2.402, 0.3), (2, 1.5), (5, 0.4), respectively. The stimulus $ws_{,s}$ was constant 1, the other states initially have value 0. The learnt connection initially is 0.1, and Δt was 1. In Fig. 2 (left hand side) a graph of the resulting state values can be seen, whereas the right hand side shows a graph of the learning. There are 19 lines, which stand for the 19 different states. However, in the graph, only 18 lines are visible. This is due to the fact that line 4 and line 16 share the same values and therefore line 16 overlaps line 4, which makes line 4 not visible in the graph. According to the scenario, the lines of the states in this graph are not fully satisfactory yet as they do not show the right sequence of events and they also do not represent the fluctuation of the states as expected. Therefore, further parameter tuning was needed to get the lines of the states as expected according to the agent model with the assigned plus and minus weights.

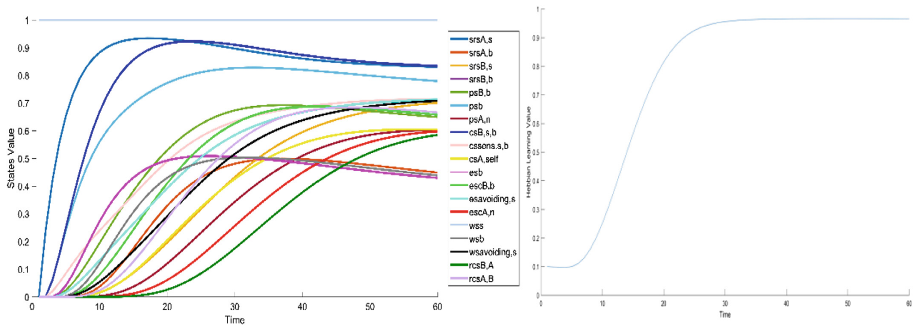

Fig. 2. A first example simulation of the agent model: states (left graph) and the adaptive connection (right graph) based on Hebbian learning

Table 2. Connection weights in the example simulations

from\to	$srs_{A,s}$	$srs_{A,b}$	$srs_{B,s}$	$srs_{B,b}$	$rsc_{B,A}$	$rsc_{A,B}$	ps_b	$ps_{B,b}$	$ps_{A,n}$	$cs_{B,s,b}$	$cs_{A,self}$	$cs_{sens,s,b}$	es_b	$esc_{B,b}$	$esc_{A,n}$	$es_{avoiding,s}$	ws_s	ws_b	$ws_{avoiding,s}$
$srs_{A,s}$							1			1		0.3							
$srs_{A,b}$								1				1							
$srs_{B,s}$											1								
$srs_{B,b}$											1								
$rsc_{B,A}$												1							
$rsc_{A,B}$											1								
ps_b												0.1	1						
$ps_{B,b}$														1					
$ps_{A,n}$															1				
$cs_{B,s,b}$								1											
$cs_{A,self}$									1										
$cs_{sens,s,b}$							-0.9						-0.5			1			
es_b				1														1	
$esc_{B,b}$						1													
$esc_{A,n}$					1														
$es_{avoiding,s}$																			1
ws_s	1																1		
ws_b		1																	
$ws_{avoiding,s}$	-0.5		1																

After more advanced tuning of the parameters (an explanation of how that was done can be found in Sect. 5), a second example simulation was obtained; see Fig. 3.

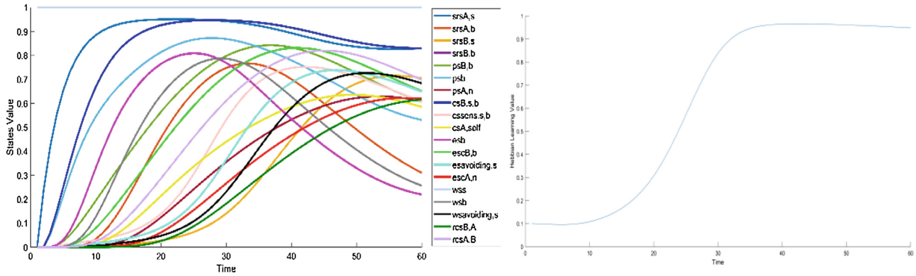

Fig. 3. Second example simulation of the agent model: states (left graph) and the adaptive connection (right graph) based on Hebbian learning

Both simulation examples show the Hebbian learning process of person A. The Hebbian learning connection is already present from the start, however initially it is low: 0.1. After person A has an interaction with Person B, Person A will learn by making the Hebbian learning connection stronger, until it is around 1. The next time person A faces an upsetting external stimulus, person A will be able to handle the external stimulus all by him or herself. In the second example simulation in Fig. 4 it can be seen that the line slowly lowers after some time, this can be explained by the fact that learning strength lowers after some time due to lower state values for ps_b and $cs_{sens,s,b}$, and therefore there is some extinction.

5 The Role of Requirements and Parameter Tuning

In this section it is described how a requirement for the expected pattern for the simulation was identified and used in automated parameter tuning based on Simulated Annealing. As mentioned in Sect. 4, the order of activations in the first example simulation was not satisfactory. Therefore, based on [15] a requirement was identified for the order of activation. This requirement was expressed as a form of use case in Table 3. Note that the notions of requirement and use case were borrowed from the Software Engineering area, and turned out useful in this case, in particular in combination with parameter tuning.

Table 3. Required succession of activation of the states: use case for the model

Time	1	2	3	4	5	6	7	8	9	10
State	ws_s	$srs_{A,s}$	$cs_{sens,s,b}$ $cs_{B,s,b}$ ps_b	es_b $ps_{B,b}$ $es_{avoiding,s}$	$esc_{B,b}$ $ws_{avoiding,s}$ ws_b $srs_{B,b}$	$rsc_{A,B}$ $srs_{A,b}$ $cs_{A,self}$ $srs_{B,s}$	$ps_{A,n}$	$esc_{A,n}$	$rsc_{B,A}$	$cs_{sens,s,b}$

This use case was expressed in activation numbers for the states that show how after some point in time from the value 0 the activation level of a state becomes high and later lower. These numbers were used as a form of pseudo-empirical data in the parameter tuning process for the speed factor parameters based on Simulated Annealing. During the tuning process the sum of squared residues SSR was used as an error function. The pattern over the number of iterations is shown in Fig. 4. The parameter values found are shown in Table 4. The graphs for these values are the ones shown in Fig. 3 above. The minimal error SSR found for these speed factors is 117.2719. The number of data points used is $N = m * n$; where m is the number of states (19) and n the number of time points used per state (60), so $N = 19 * 60 = 1140$, and SSR/N = 0.07. The square root of this is an indication for the average deviation, this is 0.26. This value could be considered a bit high; however, recall that the requirement was the focus, and the requirement has been satisfied by the tuning process. There was no further requirement that number wise the empirical data should be approximated. Therefore, by satisfying the requirement, the tuning process served our purposes.

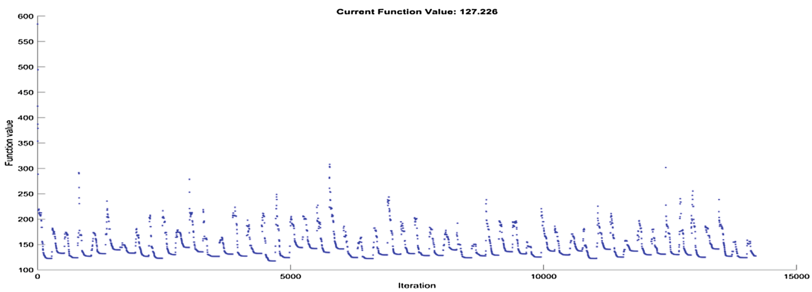

Fig. 4. Graph of the sum of squares SSR error over the number of iterations

Table 4. Speed factor values after parameter tuning

State	$srs_{A,s}$	$srs_{A,b}$	$srs_{B,s}$	$srs_{B,b}$	$rsc_{B,A}$	$rsc_{A,B}$	ps_b	$ps_{B,b}$	$ps_{A,n}$	$cs_{B,s,b}$	$cs_{A,self}$	$cs_{sens,s,b}$
Speed factor	0.359	0.5	0.498	0.368	0.412	0.474	0.5	0.469	0.393	0.493	0.493	0.498
State	es_b	$esc_{B,b}$	$esc_{A,n}$	$es_{avoiding,s}$	ws_s	ws_b	$ws_{avoiding,s}$					
Speed factor	0.5	0.473	0.473	0.452	0.498	0.5	0.487					

6 Verification of the Agent Model by Mathematical Analysis

In this section it is shown how Mathematical Analysis was used to verify whether the implemented model is in accordance with the model specification. To this end stationary points of the following states were identified and analysed: es_b, ps_b, $cs_{sens,s,b}$, $cs_{B,s,b}$ and $esc_{B,b}$. When Y is a state, Y has a stationary point at t if $\mathbf{d}Y(t)/\mathbf{d}t = 0$. For a temporal-causal network there is a simple and very effective criterion for having a stationary point: state Y has a stationary point at t ⇔ $\mathbf{aggimpact}_Y(t) = Y(t)$ ⇔ $\mathbf{c}_Y(\boldsymbol{\omega}_{X1,Y}X_1(t), \ldots, \boldsymbol{\omega}_{Xk,Y}X_k(t)) = Y(t)$ where X_i are the states with connections to state Y. To apply this, from a simulation for each of the five states state a time point was identified at which a stationary point occurred (in particular, a maximum): see Table 4, first and second row; the state values at these time points are in the third row. Next, for each of the columns, for these time points the relevant aggregated impact was calculated based on the combination function, the connection weights and the state values used in this aggregated impact: see the fourth row in Table 5. Finally, in the fifth row the absolute deviation between aggregated impact and state value is shown: $|\mathbf{aggimpact}_Y(t) - Y(t)|$.

Table 5. Verification of the model by analyzing stationary points

Maxima	es_b	ps_b	$cs_{sens,s,b}$	$cs_{B,s,b}$	$esc_{B,b}$
Time-point	25	28	42	28	42
State value	0.808715	0.872455	0.752271	0.947237	0.831066
Aggregated impact	0.803666	0.867326	0.745274	0.946509	0.826951
Absolute deviation <0.01	0.005049	0.005129	0.006997	0.000728	0.004115

As can be seen above, all five states have an absolute deviation below 0.01, which provides evidence that the implemented model does what is expected.

7 Discussion

The human-like agent model introduced in this paper was designed as a temporal-causal network based on the Network-Oriented Modeling approach described in [14]. This approach allows to model adaptive and cyclic processes based on causal relations. The network-oriented agent model designed incorporates mechanisms found in literature from Neuroscience such as [11, 12]. Such mechanisms determined the basic architecture of the network. The network model was able to reproduce the pattern what is expected from [11].

Such a model has a relatively high number of parameters. Values for such parameters cannot be obtained from the literature; they have to be estimated in one way or the other. Doing that ‘by hand’ can be a quite elaborate process. Therefore, a different approach was chosen through which there is automated support for this process. Using notions known from Software Engineering, the expected pattern was formulated as a requirement, for which a (qualitative) use case was generated. For this use case, numbers were created to make numerical methods applicable. The method used was Simulated Annealing. Indeed, the solution found satisfies the requirement: it generates the expected pattern. So, in this case this approach via requirements and use cases in combination with automated parameter tuning was quite helpful.

Sometimes it is suggested that by parameter tuning you can generate any pattern from any model if suitable parameter values are used. In general, this is not true, as long as a specific architecture is used, as in this case, based on literature [11], in order to get a human-like model. Then what is shown is that in particular this architecture, which is justified by [11], is able to simulate the expected pattern. That this architecture can also generate other patterns and that also other architectures can generate the same pattern might be true but is not relevant.

The introduced computational model can be applied as a basis for a virtual training environment for therapists to get more insight in the processes taking place in highly sensitive persons. As another step to enhance the value of such an environment, the model may be extended by a number of possible therapies and their effects on these processes.

Acknowledgement. We would like to express our gratitude to Fawad Taj for his support for the Matlab Network-Oriented Modeling template in which the model was implemented.

References

1. Jagiellowicz, J., Aron, A., Aron, E.N.: Relationship between the temperament trait of sensory processing sensitivity and emotional reactivity. Soc. Behav. Pers. Int. J. **44**(2), 185–199 (2016)
2. Aron, E.N.: High sensitivity as one source of fearfulness and shyness: preliminary research and clinical implications. In: Schmidt, L.A., Schulkin, J. (eds.) Extreme Fear, Shyness, and Social Phobia: Origins, Biological Mechanisms, and Clinical Outcomes, pp. 251–272. Oxford University Press, New York (1999)
3. Belsky, J., Pluess, M.: Beyond diathesis stress: differential susceptibility to environmental influences. Psychol. Bull. **135**(6), 885–905 (2009)
4. Aron, E.N.: Counseling the highly sensitive person. Couns. Hum. Dev. **28**(9), 1–7 (1996)
5. Aron, E.N.: The Highly Sensitive Child: Helping Our Children Thrive When the World Overwhelms Them. Harmony, New York (2002)
6. Aron, E.N.: The Highly Sensitive Person in Love. Harmony, New York (2000)
7. Rytovaara, M., Aron, E.N.: The Highly Sensitive Person. How to Thrive When the World Overwhelms You. Thorsons, London (1999)
8. Aron, E.N., Aron, A., Jagiellowicz, J.: Sensory processing sensitivity: a review in the light of the evolution of biological responsivity. Pers. Soc. Psychol. Rev. **16**(3), 262–282 (2012)
9. Aron, E.N., Aron, A.: Sensory-processing sensitivity and its relation to introversion and emotionality. J. Pers. Soc. Psychol. **73**(2), 345–368 (1997)
10. Benham, G.: The highly sensitive person: stress and physical symptom reports. Pers. Individ. Differ. **40**(7), 1433–1440 (2006)
11. Hebb, D.O.: The Organization of Behavior: A Neuropsychological Theory. Psychology Press, London (2005)
12. Munakata, Y., Pfaffly, J.: Hebbian learning and development. Dev. Sci. **7**(2), 141–148 (2004)
13. Jagiellowicz, J., Xu, X., Aron, A., Aron, E.N., Cao, G., Feng, T., et al.: The trait of sensory processing sensitivity and neural responses to changes in visual scenes. Soc. Cogn. Affect. Neurosci. **6**(1), 38–47 (2011)
14. Kim, J.: Philosophy of Mind. Westview Press, Boulder (1996)
15. Treur, J.: Network-Oriented Modeling: Addressing Complexity of Cognitive, Affective and Social Interactions. Springer, Cham (2016). https://doi.org/10.1007/978-3-319-45213-5

16. Kuipers, B.J.: Commonsense reasoning about causality: deriving behavior from structure. Artif. Intell. **24**, 169–203 (1984)
17. Kuipers, B.J., Kassirer, J.P.: How to discover a knowledge representation for causal reasoning by studying an expert physician. In: Proceedings of the 8th International Joint Conference on Artificial Intelligence, IJCAI 1983, pp. 49–56. William Kaufman, Los Altos (1983)
18. Pearl, J.: Causality. Cambridge University Press, Cambridge (2000)